P- 3466

IMPLEMENTATION OF FIXED WIDTH PARALLEL AND TRUNCATED MULTIPLIERS

By

ANU JOSE

Reg. No.1020106001

of

KUMARAGURU COLLEGE OF TECHNOLOGY

(An Autonomous Institution affiliated to Anna University of Technology, Coimbatore)

COIMBATORE - 641049

A MINI PROJECT REPORT

Submitted to the

FACULTY OF ELECTRONICS AND COMMUNICATION ENGINEERING

In partial fulfillment of the requirements

for the award of the degree

of

MASTER OF ENGINEERING

IN

APPLIED ELECTRONICS

MAY 2011

BONAFIDE CERTIFICATE

Certified that this project report entitled "**IMPLEMENTATION OF FIXED WIDTH PARALLEL AND TRUNCATED MULTIPLIERS** "is the bonafide work of **Ms. ANU JOSE [Reg no: 1020106001]** who carried out the mini project under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

Project Guide

Ms.G Amirtha Gowri

for **Head of the Department**

Dr. Rajeswari Mariappan

The candidate with university Register no. 1020106001 is examined by us in the mini project viva-voce examination held on ...23-05-11....

Internal Examiner

External Examiner

ACKNOWLEDGEMENT

A project of this nature needs co-operation and support from many for successful completion. In this regards, I am fortunate to express my heartfelt thanks to Chairman **Padmabhusan Arutselvar Dr.N.Mahalingam B.Sc.,F.I.E.,** and Co-Chairman **Dr.B.K.Krishnaraj Vanavarayar B.Com,B.L.,** for providing necessary facilities throughout the course.

I would like to express my thanks and appreciation to the many people who have contributed to the successful completion of this project. First I thank, **Dr.J.Shanmugam, Ph.D,** Director, for providing me an opportunity to carry out this project work.

I would like to thank **Dr.S.Ramachandran, Ph.D,** Principal, who gave her continual support and opportunity for completing the project work successfully.

I would like to thank **Dr.Rajeswari Mariappan, Ph.D,** Prof of Head, Department of Electronics and Communication Engineering, who gave her continual support for us throughout the course of study.

I would like to thank **Ms.G,Amirtha Gowri M.E(Ph.D),** Assistant Professor, Project guide for her technical guidance, constructive criticism and many valuable suggestions provided throughout the project work.

My heartfelt thanks to **Ms.R.Hemalatha,** Associate Professor, Project coordinator, for her contribution and innovative ideas at various stages of the project to successfully complete this work.

I express my sincere gratitude to my family members, friends and to all my staff members of Electronics and Communication Engineering Department for their support throughout the course of my project.

ABSTRACT

This paper presents a comparative study of Field Programmable Gate Array (FPGA) implementation of standard and truncated multipliers using Very High Speed Integrated Circuit Hardware Description Language (VHDL). Truncated multiplier is a good candidate for digital signal processing (DSP) applications such as finite impulse response (FIR) and discrete cosine transform (DCT) etc. Significant reduction in FPGA resources, delay, and power can be achieved using truncated multipliers instead of standard parallel multipliers when the full precision of the standard multiplier is not required. The power and area of truncated 6×6-bit multiplier shows significant improvement as compared to standard 6×6-bit multiplier. The truncated multiplier shows a reduction in power and area by 60% and 34% respectively as compared to standard multiplier.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Multiplication is the core operation in many signal processing algorithms including: convolution, Euclidean distance, filtering and Fast Fourier Transform (FFT). The computational complexities of algorithms used in digital signal processing (DSP) have steadily increased. As a result, fast and efficient parallel multipliers are required for general purpose digital signal processors (DSPs) as well as application specific architectures for DSP. In particular, if the processing has to be performed under real time conditions, such algorithms have to deal with high throughput rates. In many cases implementation of DSP algorithm demands using application specific integrated circuits (ASICs). This is especially required for image processing applications. Since development costs for ASICs are high, algorithms should be verified and optimized before implementation.

Recent advancements in VLSI technology and in particular, the increasing complexity and capacity of state-of-the-art programmable logic devices have been making hardware emulations possible. The underlying key of the emulation system is to use SRAM-based field programmable gate arrays (FPGAs) which are very flexible and dynamically reconfigurable. In FPGAs, the choice of the optimum multiplier involves three key factors: area, propagation delay and reconfiguration time.

1.1. PROJECT GOAL

Reducing the power dissipation of parallel multipliers is important in the design of digital signal processing systems. In many of these systems, the products of parallel multipliers are rounded to avoid growth in word size. The power dissipation and area of rounded parallel multipliers can be significantly reduced by a technique known as truncated multiplication. With this technique, the least significant columns of the multiplication matrix are not used. Instead, the carries generated by these columns are estimated. This estimate is added with the most significant columns to produce the

rounded product. This project aims the design and implementation of parallel truncated multipliers. ..

1.2 OVERVIEW

Multiplication and squaring functions are used extensively in applications such as DSP, image processing and multimedia. A full width digital n×n multiplier computes the 2n output as a weighted sum of partial products. If the product is truncated to n-bits, the least-significant columns of the product matrix contribute little to the final result. To take advantage of this, truncated multipliers and squarers do not form all of the least-significant columns in the partial-product matrix. As more columns are eliminated, the area and power consumption of the arithmetic unit are significantly reduced, and in many cases the delay also decreases. The trade-off is that truncating the multiplier matrix introduces additional error into the computation.

1.3 SOFTWARE USED

- ModelSim XE 111 6.3f
- Xilinx ISE 8.1i

1.4 ORGANIZATION OF THE REPORT

- **Chapter 2** Discusses the algorithm for standard parallel multiplier
- **Chapter 3** Discusses the algorithm for truncated multiplier
- **Chapter 4** Discusses the algorithm for floating point multiplication.
- **Chapter 5** Discusses floating point representation
- **Chapter 6** Discusses results and discussion.
- **Chapter 7** Discusses merits, demerits and applications of truncated multiplier
- **Chapter 8** Discusses conclusion.

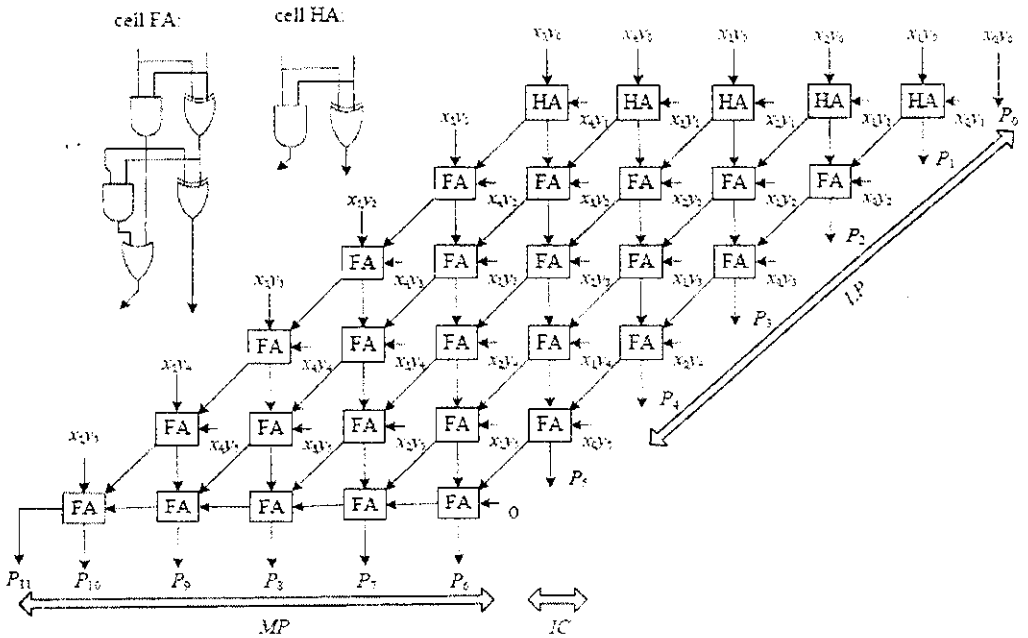
CHAPTER 2

ALGORITHM FOR STANDARD PARALLEL MULTIPLIER

Considering the multiplication of two n-bit inputs X and Y, a standard multiplier performs the following operations to obtain the $2n$ bit product P

$$P = XY = \sum_{i=0}^{2n-1} P_i 2^i = (\sum_{i=0}^{n-1} x_i 2^i)(\sum_{i=0}^{n-1} y_i 2^i) \quad (1)$$

where xi, yi and Pi represent the ith bit of X, Y and P, respectively. Fig1 shows the standard architecture of 6×6-bit parallel multiplier, where HA and FA are the half and full adders respectively

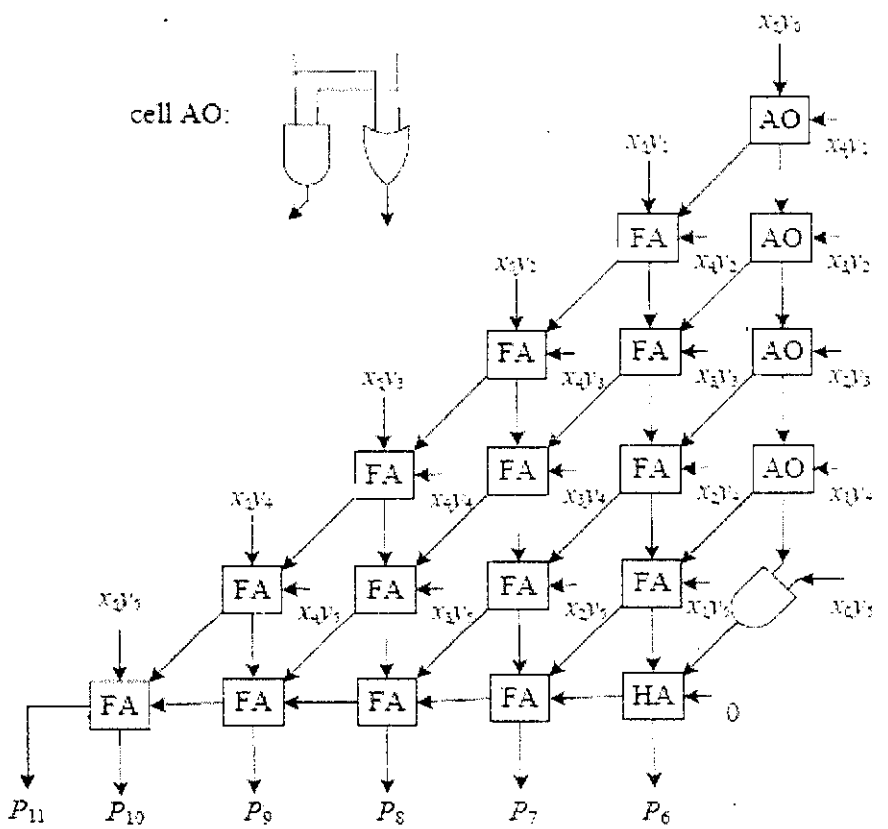


CHAPTER 3

ALGORITHM FOR TRUNCATED MULTIPLIER

Truncated multipliers compute the n most-significant bits of the n × n bits product.The fixed width multiplier can be obtained directly by removing the *LP* region and introducing the *IC* region to obtain *MP* region, which is truncated multiplier as shown in Fig. 2 and given by equation (4).

$$P = MP' + IC \tag{4}$$



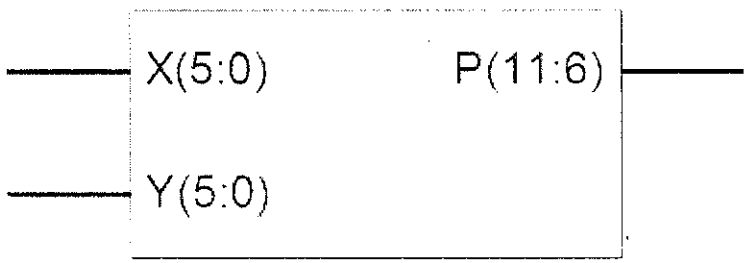


Figure 3.1: Block diagram of 6*6 truncated multiplier

CHAPTER 4

ALGORITHM FOR FLOATING POINT MULTIPLICATION

This project has been extended for floating point numbers too. The algorithm for the same is as below:

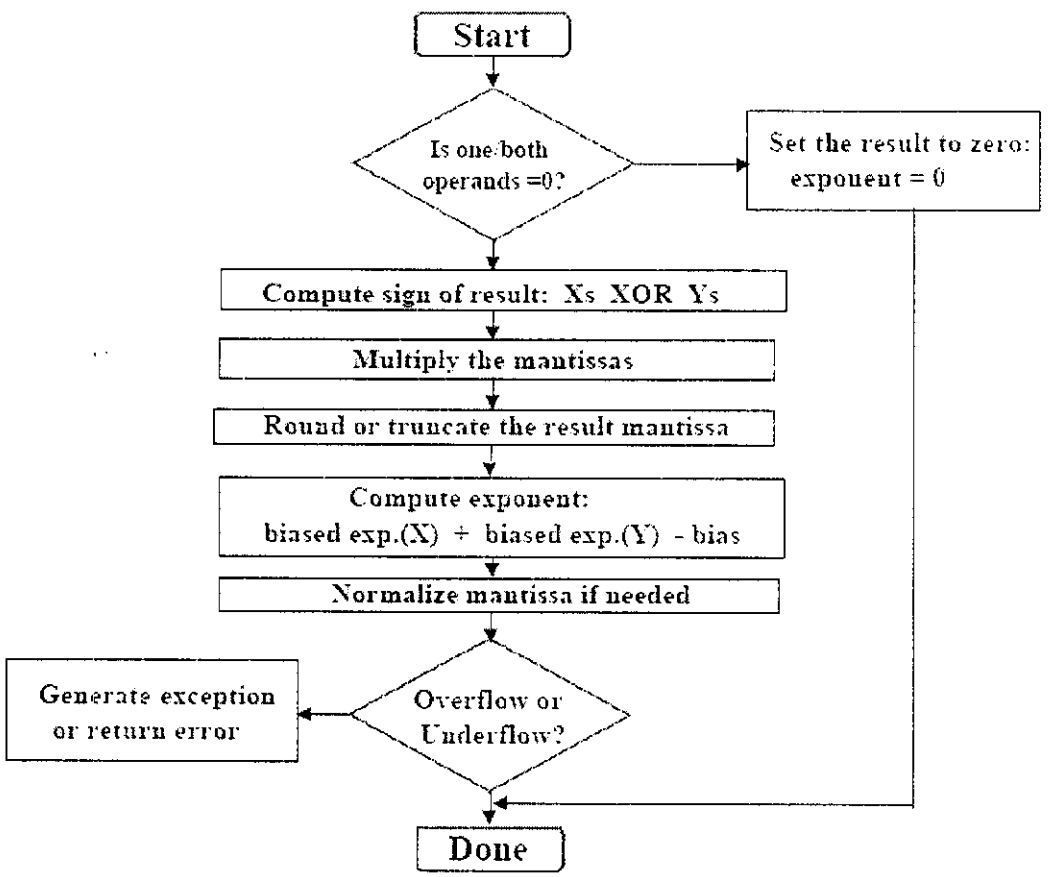


Figure 4.1:Flow chart for floating point multiplication.

CHAPTER 5

FLOATING POINT REPRESENTATION

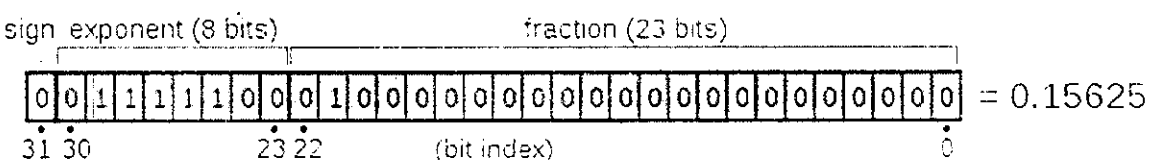
5.1 IEEE 754 SINGLE PRECISION BINARY FLOATING-POINT FORMAT: BINARY32

The IEEE 754 standard specifies a **binary32** as having:

- Sign bit:1 bit
- Exponent width: 8 bits
- Significant precision: 24 (23 explicitly stored)

Sign bit determines the sign of the number, which is the sign of the significant as well. Exponent is either an 8 bit signed integer from −128 to 127 or an 8 bit unsigned integer from 0 to 255 which is the accepted biased form in IEEE 754 binary32 definition. For this case an exponent value of 127 represents the actual zero.

The true significant includes 23 fraction bits to the right of the binary point and an implicit leading bit (to the left of the binary point) with value 1 unless the exponent is stored with all zeros. Thus only 23 fraction bits of the significand appear in the memory format but the total precision is 24 bits (equivalent to $\log_{10}(2^{24}) \approx 7.225$ decimal digits). The bits are laid out as follows:



5.2 FLOATING POINT MULTIPLICATION EXAMPLE

Multiply the following two numbers represented in the IEEE 754 single precision format: X = -1.2 and Y = 1.4

X = -1.2

1	**011111111**	**001100**

Y = 1.4

0	**01111111**	**011001**

1. Compute the sign: S = Xs XOR Ys = 1 XOR 0 = 1.
2. Multiply the mantissas:

   Xm = 001100 Ym = 011001 OUTm = 101011
3. Compute exponent of result:

   Xe + Ye – 127(10) = 01111111

CHAPTER 6

RESULTS AND DISCUSSION

The design of parallel and truncated 6*6 multiplier for integer and floating point values are done using VHDL and are simulated in MODEL SIM XE 111 6.3f.Using the XILINX ISE 8.1i software the power consumption, area and delay of both the multipliers have been compared. For a truncated multiplier the power consumption is decreased by 60%, area is decreased by 30% and delay is decreased by 6% approximately. The comparisons are given in the table below.

PARAMETERS	PARALLELMULTIPLIER	TRUNCATEDMULTIPLIER
POWER CONSUMPTION	137mW	55mW
GATE COUNT	402	264
PATH DELAY	21.966ns	20.576ns

The power, delay and map report and the various simulation results are included below.

6.1:STANDARD PARALLEL MULTIPLIER:

POWER REPORT:

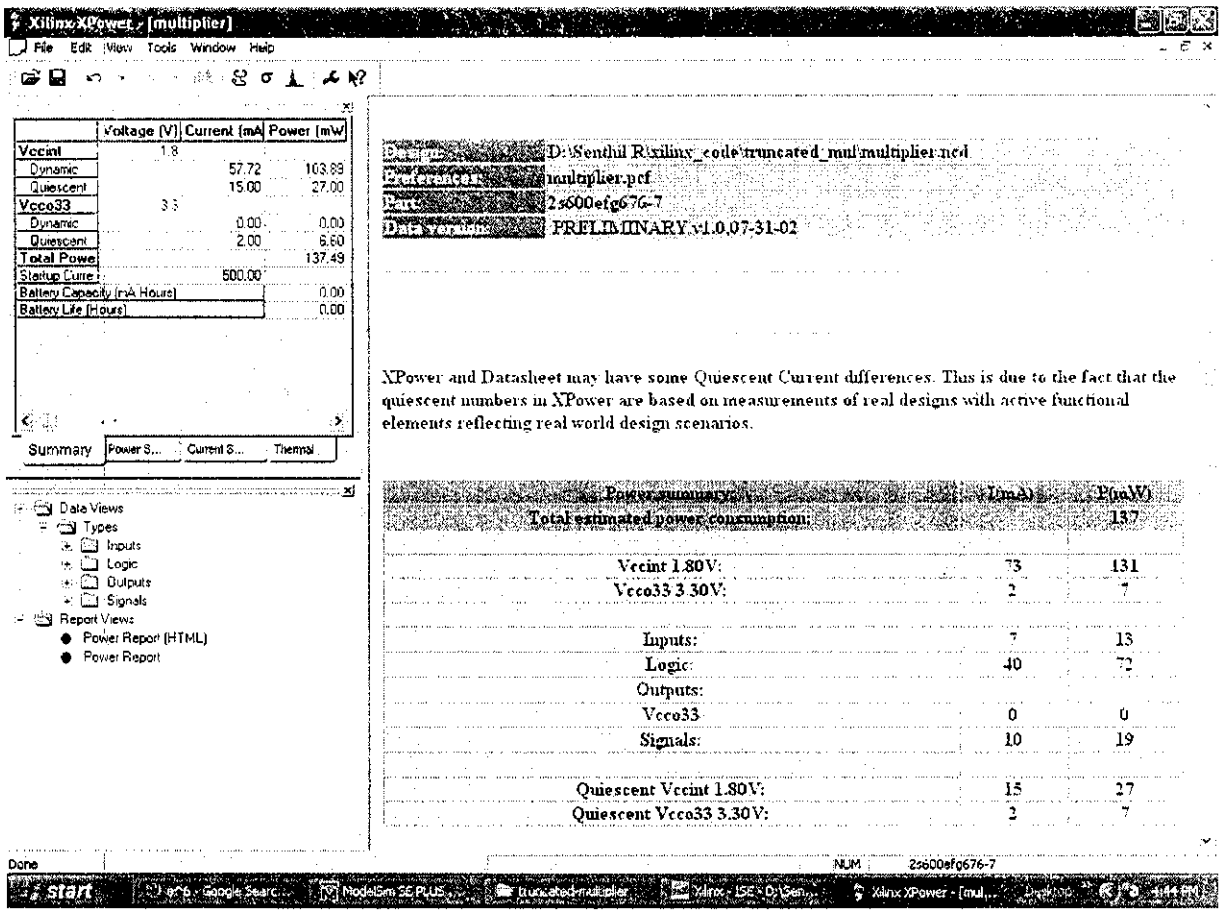


Figure 6.1:Power report for parallel multiplier

P-3466

MAP REPORT:

Design Summary

Logic Utilization:

Number of 4 input LUTs: 67 out of 13,824 1%

Logic Distribution:

Number of occupied Slices: 35 out of 6,912 1%

Number of Slices containing only related logic: 35 out of 35 100%

Number of Slices containing unrelated logic: 0 out of 35 0%

Total Number of 4 input LUTs: 67 out of 13,824 1%

Number of bonded IOBs: 24 out of 510 4%

Total equivalent gate count for design: 402

Additional JTAG gate count for IOBs: 1,152

DELAY REPORT:

Timing Summary:

Minimum period: No path found

Minimum input arrival time before clock: No path found

Maximum output required time after clock: No path found

Maximum combinational path delay: 21.966ns

6.2:TRUNCATED MULTIPLIER:

POWER REPORT:

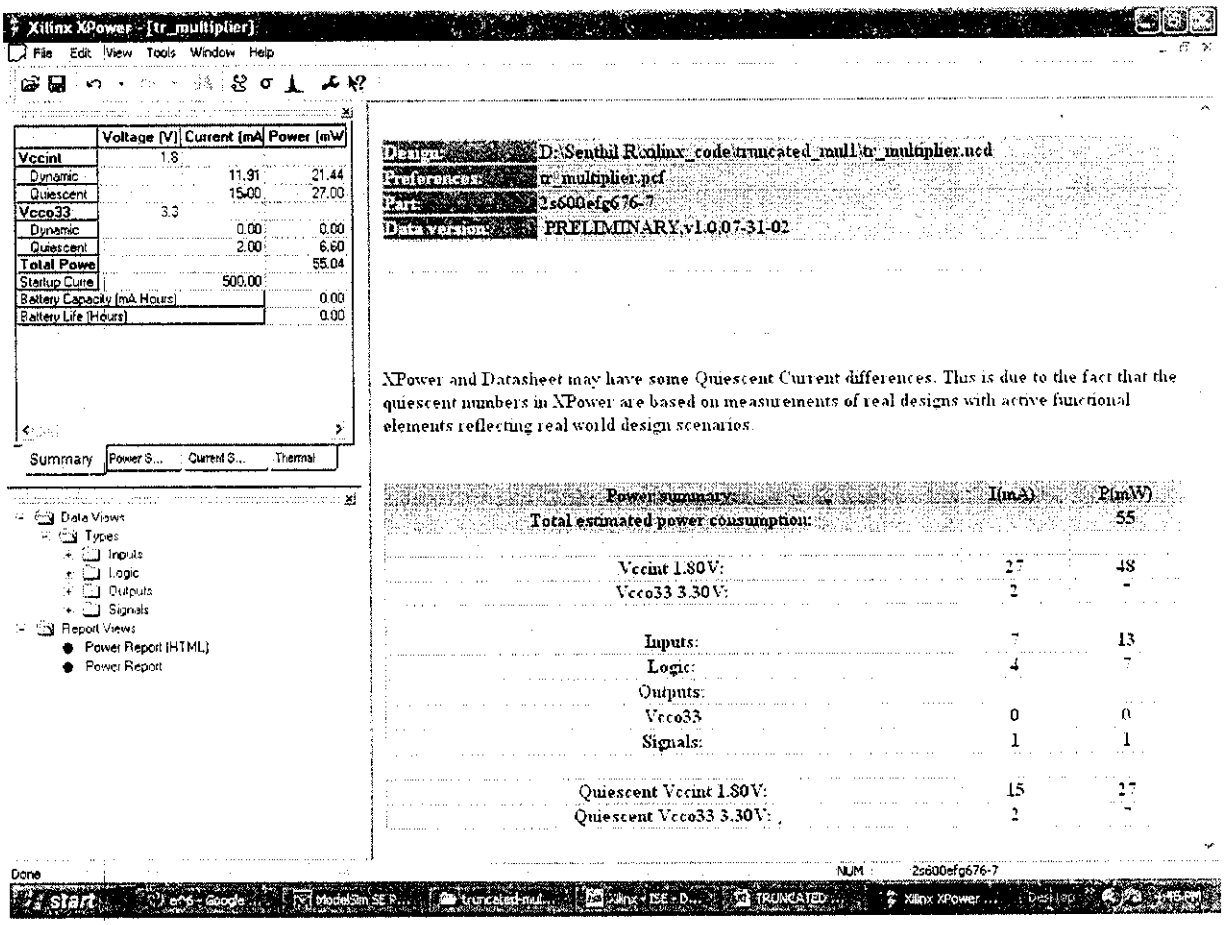


Figure 6.2:Power report for truncated multiplier.

MAP REPORT:

Design Summary

Logic Utilization:

Number of 4 input LUTs: 44 out of 13,824 1%

Logic Distribution:

Number of occupied Slices: 24 out of 6,912 1%

Number of Slices containing only related logic: 24 out of 24 100%

Number of Slices containing unrelated logic: 0 out of 24 0%

Total Number of 4 input LUTs: 44 out of 13,824 1%

Number of bonded IOBs: 18 out of 510 3%

Total equivalent gate count for design: 264

Additional JTAG gate count for IOBs: 864

DELAY REPORT:

Timing Summary:

Minimum period: No path found

Minimum input arrival time before clock: No path found

Maximum output required time after clock: No path found

Maximum combinational path delay: 20.576ns

6.3:STANDARD PARALLEL MULTIPLIER

FOR INTEGER VALUES

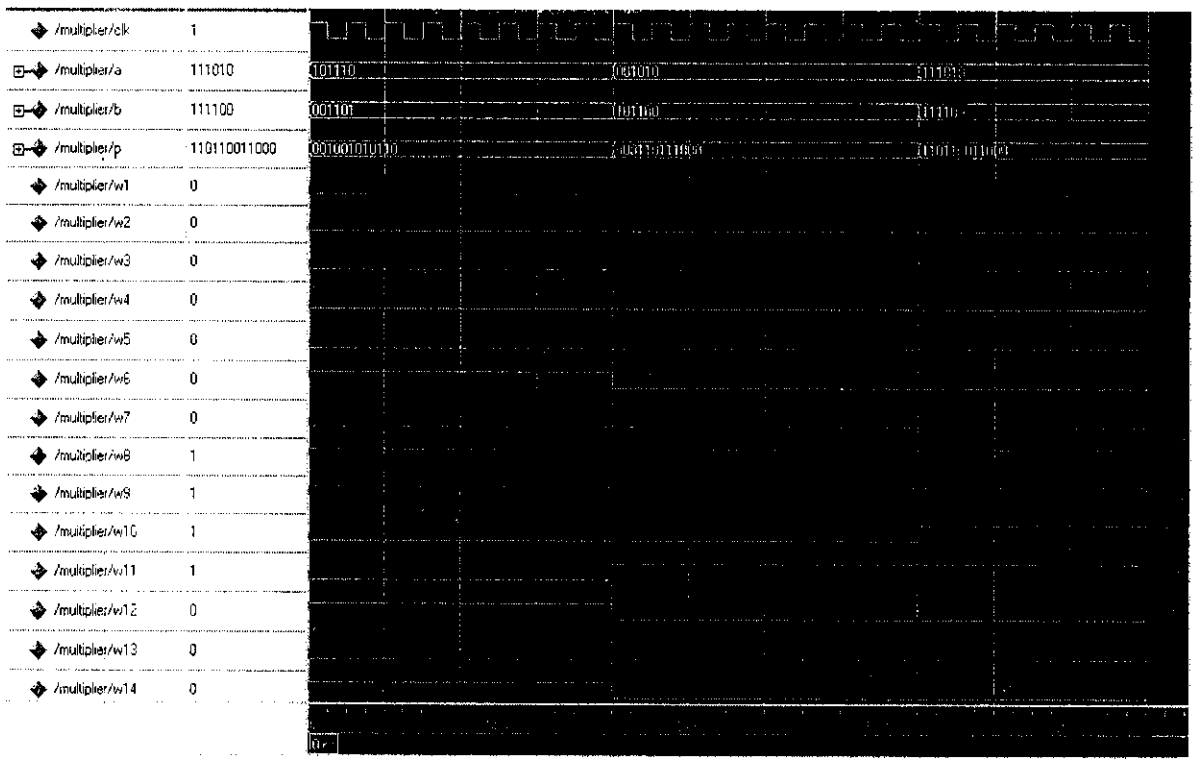


Figure 6.3.Simulation result of parallel multiplier for integer values

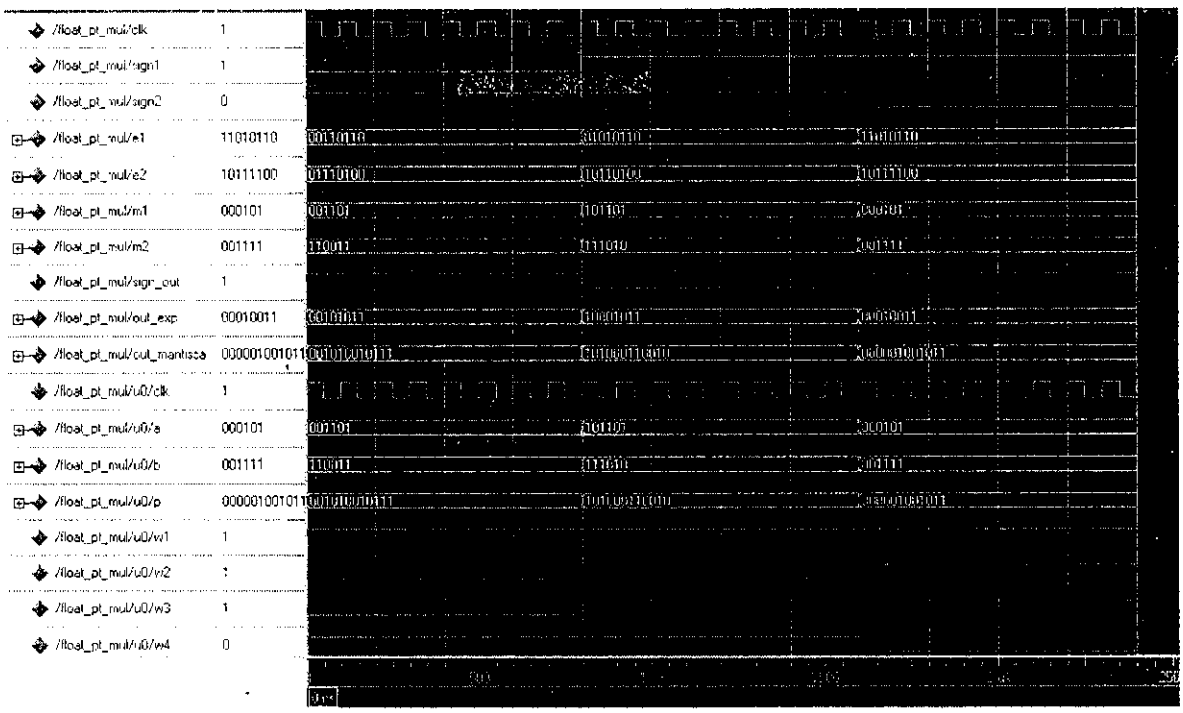


Figure6.4.Simulation result of parallel multiplier using floating point arithmetic.

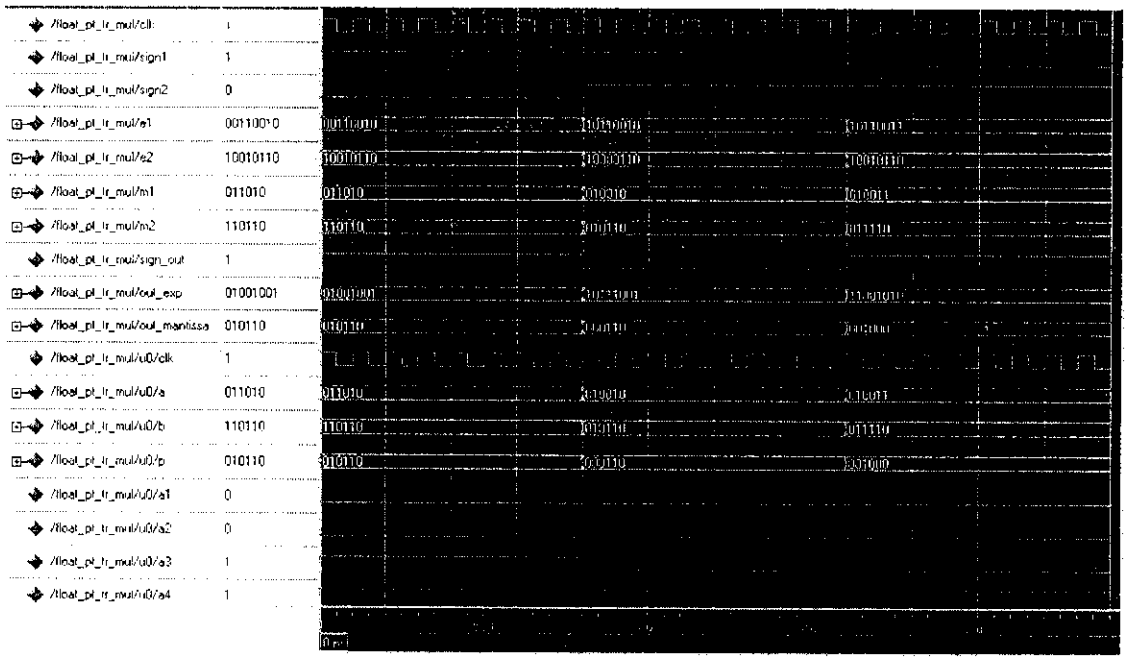


Figure 6.6:Simulation result of truncated multiplier for floating point arithmetic.

CHAPTER 7

MERITS, DEMERITS AND APPLICATION OF TRUNCATED MULTIPLIER

MERITS:

- Area of the arithmetic unit reduces
- Power consumption decreases.
- Delay decreases
- Reduction in device utilization

DEMERITS:

- Truncating the multiplier matrix introduces additional error in to the computation.

APPLICATIONS:

- Used in applications such as DSP, image processing and multimedia.

CHAPTER 8

CONCLUSION

In this project, an FPGA implementation of standard and truncated 6×6-bit multiplier utilizing VHDL has been done..Both the design were implemented on Xilinx Spartan 2E FPGA device. The objective is to present a comparative study of the standard and truncated multipliers. The truncated multiplier shows much more reduction in device utilization as compared to standard multiplier. The presented FPGA implementation of truncated multiplier achieves a power consumption of 55 mW that is the power consumption is reduced by 60%.

BIBLIOGRAPHY

- C. Maxfield, *The Design Warrior's Guide to FPGAs: Devices Tools and flows.* Newnes Publishers, MA: 2004.
- E. III. Walters, M. G. Arnold, and M. J. Schulte, "Using truncated multipliers in DCT and IDCT hardware accelerators,"in *Proc. XIII SPIE Advanced Signal Processing Algorithms,Architectures, and Implementations*, San Diego, California,2003, pp. 573-584.
- M. –H. Sheu and S. -H. Lin, "Fast compensative design approach of the approximate squaring function," *IEEE J. SolidState Circuits*, vol. 37, no. 1, pp. 95-97, Jan. 2002.
- C. R. Baugh and B. A. Wooley, "A Two's Complement Parallel Array Multiplication Algorithm," *IEEE Trans. Computers*, vol.C-22, no. 12, pp. 1045-1047, Dec. 1973.
- E. E. Swartzlander Jr., "Truncated Multiplication with Approximate Rounding," in *Proc. 33rd Asilomar Conference onSignals, Systems, and Computers*, vol. 2, 1999, pp. 1480-1483.
- J. E. Stine and O. M. Duverne, "Variations on Truncated Multiplication," in *Proc. Euromicro Symposium on Digital System Design*, 2003, pp. 112-119